A Bregman alternating direction method of multipliers for sparse probabilistic Boolean network problem

Kangkang Deng *, Zheng Peng *†

Abstract: The main task of genetic regulatory networks is to construct a sparse probabilistic Boolean network (PBN) based on a given transition-probability matrix and a set of Boolean networks (BNs). In this paper, a Bregman alternating direction method of multipliers (BADMM) is proposed to solve the minimization problem raised in PBN. All the customized subproblem-solvers of the BADMM do not involve matrix multiplication, consequently the proposed method is in a position to deal with some huge-scale problems. The convergence to stationary point of the BADMM is proved under some mild conditions. Numerical experiments show that the BADMM is effective and efficient comparing with some existing methods.

Keywords: Genetic regulatory networks, Sparse probabilistic Boolean network, $L_{\frac{1}{2}}$-regularization, Separable minimization, Bregman alternating direction method of multipliers.

1 Introduction

Probabilistic Boolean Network (PBN) is a rule-based uncertainty model for gene regulatory networks, see [1, 2, 3, 4, 5] and the references therein. This model is devoted to construct the probabilistic Boolean network from a given transition-probability matrix P and a set of Boolean networks (BNs) with matrices $\{P_i\}_{i=1}^n$. Mathematically, a sparse PBN problem is to find a sparse probability distribution vector $x \in R^n$, such that

$$\sum_{i=1}^{n} P_i x_i = P, \tag{1.1a}$$

$$\mathbf{e}^T x = 1, \quad x \geq 0, \tag{1.1b}$$

where $\mathbf{e} = (1, 1, ..., 1)^T$.

This problem is essentially a linear system restricted to $x \geq 0$. In system (1.1), the transition-probability matrix P and the Boolean matrices P_i are sparse. If BN has j genes, there are 2^j possible states. Correspondingly, the matrices P and $P_i (i = 1, ..., n)$ are 2^j by 2^j matrices, each column of P matrix has $l(\approx j)$ non-zero entries, and each column of P_i has a unique non-zero entry. On the other hand, system (1.1) is very large scale, in the case that there are j genes and each column of P matrix has l non-zero entries, system (1.1) has $n = l^{2^j}$ unknowns. Hence it is also indeterminate. The sparse PBN model is introduced to find a sparse solution of system (1.1) since it needs a sparse probability distribution vector.

*College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350116, China.
†Corresponding author. Email: pzheng@fzu.edu.cn.

A least squares solution of system (1.1) is given by the following minimization problem

$$\begin{cases} \min\limits_{x} \dfrac{1}{2}\Big\|P - \sum\limits_{i=1}^{n} P_i x_i\Big\|_{\mathsf{F}}^2, \\ \text{s.t. } \mathbf{e}^T x = 1,\ x \geq 0. \end{cases} \tag{1.2}$$

Let "vec" be a mapping from $R^{s\times s}$ to R^{s^2}, where

$$\text{vec}\left(\begin{pmatrix} a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & \vdots \\ a_{s1} & \cdots & a_{ss} \end{pmatrix}\right) = (a_{11}, \cdots, a_{s1}, a_{12}, \cdots, a_{s2}, \cdots, \cdots, a_{s1}, \cdots, a_{ss})^T,$$

$A = \big[\text{vec}(P_1), \text{vec}(P_2), \cdots, \text{vec}(P_n)\big]$ and $b = \text{vec}(P) \in R^m$ with $m = 2^{2j}$. Then the minimization problem (1.2) can be expressed as

$$\begin{cases} \min\limits_{x} \dfrac{1}{2}\Big\|Ax - b\|^2, \\ \text{s.t. } \mathbf{e}^T x = 1,\ x \geq 0, \end{cases} \tag{1.3}$$

which is a convex minimization problem. The coefficient matrix A is a $2^{2j} \times l^{2^j}$ matrix.

Problem (1.3) may have too many solutions. To narrow down the solution set, Chen [1] considered the solution given by the largest entropy based on the fact that x is a probability distribution. They converted problem (1.3) to

$$\begin{cases} \min\limits_{x\in R^n} \dfrac{1}{2}\Big\|Ax - b\|^2 + x^T \log x, \\ \text{s.t. } \mathbf{e}^T x = 1,\ x \geq 0. \end{cases} \tag{1.4}$$

However, to characterize the sparsity property of problem (1.3), the L_0-regularization is a common choice. By L_0-regularization we get

$$\begin{cases} \min\limits_{x\in R^n} \dfrac{1}{2}\Big\|Ax - b\|^2 + \tau\|x\|_0, \\ \text{s.t. } \mathbf{e}^T x = 1,\ x \geq 0, \end{cases} \tag{1.5}$$

where $\tau > 0$ is a regularization parameter. Problem (1.5) is NP-hard [6] and intractable. To overcome this difficulty, many researchers have relaxed $\|x\|_0$ to $\|x\|_1$. By L_1-regulation, problem (1.5) becomes

$$\begin{cases} \min\limits_{x\in R^n} \quad \dfrac{1}{2}\|Ax - b\|^2 + \tau\|x\|_1, \\ \quad s.t. \quad \mathbf{e}^T x = 1, \quad x \geq 0. \end{cases} \tag{1.6}$$

Problem (1.6) is a convex minimization problem. Many methods correlated with augmented Lagrangian-based alternating direction methods of multipliers [7, 8, 9, 10, 11, 12, 13] can be used to solve this kind of problem. These methods are globally convergent to a global solution. However, since $x \geq 0$, we have $\|x\|_1 = \sum_{i=1}^n |x_i| = \mathbf{e}^T x = 1$. The regularization term $\tau\|x\|_1$ is invalid in the constrained optimization problem (1.6).

Xu, et al [14] replaced L_0 by $L_{\frac{1}{2}}$ regularization for characterizing the sparsity. By a phase diagram study, Xu et al [15] showed that the L_q $(0 < q < 1)$ regularization can assuredly generate more sparse solutions than L_1 regularization, in which the index $q = \frac{1}{2}$ somehow

plays a representative role. If $q \in [\frac{1}{2}, 1)$, the smaller q will lead to the solution yielded by L_q regularization more sparser. If $q \in (0, \frac{1}{2}]$, the performance of L_q regularization has no significant difference. What may be the most important from the computational point of view, there is an iterative half thresholding algorithm for the fast solution of $L_{\frac{1}{2}}$ regularization.

For the $L_{\frac{1}{2}}$ regularization problem of the form

$$\min_{x \in R^n} \|Ax - b\|^2 + \gamma \|x\|_{\frac{1}{2}}^{\frac{1}{2}}, \tag{1.7}$$

the iterative half thresholding algorithm is given by [14]:

$$x^{k+1} = H_{\gamma\eta, \frac{1}{2}}(x^k + \eta A^T(b - Ax^k)), \tag{1.8}$$

where $\eta > 0$. The operator $H_{\gamma\eta, \frac{1}{2}}$ is specified by

$$H_{\gamma\eta, \frac{1}{2}}(z) = (h_{\gamma\eta, \frac{1}{2}}(x_1), h_{\gamma\eta, \frac{1}{2}}(x_2), \cdots, h_{\gamma\eta, \frac{1}{2}}(x_n))^T, \tag{1.9}$$

where

$$h_{\gamma\eta, \frac{1}{2}}(z) = \begin{cases} f_{\gamma\eta, \frac{1}{2}}(z), & |z| > \frac{\sqrt[3]{54}}{4}(\gamma\eta)^{\frac{2}{3}}, \\ 0, & \text{otherwise}, \end{cases} \tag{1.10}$$

in which

$$f_{\gamma\eta, \frac{1}{2}}(z) = \frac{2}{3} z \left(1 - \cos\left(\frac{2\pi}{3} - \frac{2}{3}\varphi_{\gamma\eta}(z)\right)\right) \tag{1.11}$$

and

$$\varphi_{\gamma\eta}(z) = \arccos\left(\frac{\gamma\eta}{8}\left(\frac{|z|}{3}\right)^{-\frac{3}{2}}\right). \tag{1.12}$$

Under some conditions, Zeng, Lin, Wang and Xu [16] have proven that the half algorithm converges to a local minimizer of the regularization problem (1.7) with an eventually linear convergence rate.

Based on the above consideration, the $L_{\frac{1}{2}}$ is used to instead of L_0 regularization in this paper. Then, we get the following $L_{\frac{1}{2}}$ regularization problem

$$\begin{cases} \min\limits_{x \in R^n} & \frac{1}{2}\|Ax - b\|^2 + \tau \|x\|_{1/2}^{1/2}, \\ s.t. & \mathbf{e}^T x = 1, \quad x \geq 0. \end{cases} \tag{1.13}$$

By using a penalty method to problem (1.13), we have

$$\min_{x \geq 0} \frac{1}{2}\|Ax - b\|^2 + \frac{1}{2\mu}(e^T x - 1)^2 + \tau \|x\|_{1/2}^{1/2}, \tag{1.14}$$

where $\mu > 0$ is a penalty parameter. A separable reformulation of (1.14) is given by

$$\begin{cases} \min & \frac{1}{2}\|Ax - b\|^2 + \frac{1}{2\mu}(e^T x - 1)^2 + \tau \|y\|_{1/2}^{1/2}, \\ s.t. & x - y = 0, \quad x \geq 0. \end{cases} \tag{1.15}$$

In this paper, we will propose a Bregman alternating direction method of multipliers (BAD-MM) to solve problem (1.15), and prove the convergence to stationary point of the proposed method.

The rest of this paper is organized as follows. In Section 2, we propose a Bregman alternating direction method of multipliers (BADMM) for problem (1.15) raised in the sparse PBN, and customize all the subproblem-solvers of the proposed method. Section 3 proves the convergence to stationary point of the proposed method. Some numerical results are presented in Section 4 to demonstrate that the proposed method is effective and efficient. Section 5 concludes this paper with some final remarks.

2 The proposed method

In this section, a Bregman alternating direction method of multipliers for problem (1.15) will be proposed, and the iteration subproblem-solvers will be customized.

The augmented Lagrangian function associated with problem (1.15) is

$$L_\rho(x,y,\lambda) \quad = \frac{1}{2}\|Ax-b\|^2 + \frac{1}{2\mu}(e^Tx-1)^2 + \tau\|y\|_{\frac{1}{2}}^{\frac{1}{2}} - \lambda^T(x-y) + \frac{\rho}{2}\|x-y\|^2, \tag{2.1}$$

where $\lambda \in R^n$ is a Lagrange multiplier and $\rho > 0$ is a penalty parameter.

Based on the augmented Lagrangian function, we propose a Bregman alternating direction method of multipliers to solve problem (1.15) as follows.

Algorithm 2.1. Bregman alternating direction method of multipliers, BADMM
For a given (x^k, y^k, λ^k), the new iterate $(x^{k+1}, y^{k+1}, \lambda^{k+1})$ is generated via:

$$y^{k+1} = \mathrm{Arg}\min_{y\geq 0}\Big\{L_\rho(x^k,y,\lambda^k)\Big\}, \tag{2.2}$$

$$x^{k+1} = \mathrm{Arg}\min_{x}\Big\{L_\rho(x,y^{k+1},\lambda^k) + \frac{r}{2}\|x-x^k\|^2,\ r>0\Big\}, \tag{2.3}$$

$$\lambda^{k+1} = \lambda^k - \rho(x^{k+1}-y^{k+1}). \tag{2.4}$$

For subproblem (2.2) we have

$$y^{k+1} = \mathrm{Arg}\min_{y\geq 0}\Big\{\tau\|y\|_{1/2}^{1/2} + \frac{\rho}{2}\|y-(x^k+\frac{1}{\rho}\lambda^k)\|^2\Big\}. \tag{2.5}$$

Let $\epsilon \in (0,1)$ be a small number, and $\eta = 1-\epsilon$. Then, an iterative half thresholding algorithm for problem (2.5) can be described as follows.

Algorithm 2.2. Iterative half thresholding algorithm for subproblem (2.5)

s0. initial: $\tilde{y}^0 = y^k$, $t = 0$, $et = 1.0$;

s1. while $et > \epsilon$, do

$$\theta(\tilde{y}^t) = y^t - \eta(y^t - (x^k - \frac{1}{\rho}\lambda^k));$$

$$\kappa = \frac{\sqrt{96}}{9\eta}\left|[\theta(\tilde{y}^t)]_{s+1}\right|^{\frac{3}{2}}$$

where $[v]_r$ denotes the $r-$th largest component of v.

compute a new predictor via

$$\tilde{y}^{t+1} = H_{\kappa\eta,\frac{1}{2}}(\theta(\tilde{y}^t))$$

update $et = \|\tilde{y}^{t+1} - \tilde{y}^t\|_\infty$;

end (while)

s2. compute a new iterate via

$$y^{k+1} = \max\{\tilde{y}^{t+1}, 0\}.$$

The iterative half thresholding algorithm is computationally tractable. Hence, the subproblem (2.2) is referred to as an easy problem.

Now, we turn to subproblem (2.3). Note that

$$L_\rho(x, y^{k+1}, \lambda^k) = \frac{1}{2}\|Ax-b\|^2+\frac{1}{2\mu}(e^T x-1)^2-(\lambda^k)^T(x-y^{k+1})+\frac{\rho}{2}\|x-y^{k+1}\|^2+\tau\|y^{k+1}\|_{\frac{1}{2}}^{\frac{1}{2}}. \quad (2.6)$$

Since A matrix is huge size, the storage-cost of full matrix A may be very expensive. By the sparsity of $A = [f_1, \cdots, f_n]$ where $f_i = \mathrm{vec}(P_i)$, a coordinate descent method is recommended to solve subproblem (2.3).

Let

$$F_k(x) = L_\rho(x, y^{k+1}, \lambda^k) + \frac{r}{2}\|x - x^k\|^2.$$

The coordinate descent method for subproblem (2.3) can be stated as follows:

Algorithm 2.3. Coordinate descent method for subproblem (2.3).

s0. initial: let $\tilde{x}^0 = x^k$, $l = 0$, $el = 1.0$;

s1. while $el > \epsilon$, do

for $i = 1, 2, \cdots, n$, let

$$\tilde{x}_i^{1+1} = \arg\min F_k(\tilde{x}_{i-}^{l+1}, x_i, \tilde{x}_{i+}^{l})$$

end (for)

update $el = \|\nabla F_k(\tilde{x}^{l+1})\|$;

end (while)

s2. $x^{k+1} = \tilde{x}^{l+1}$

By the definition, $f_i = \mathrm{vec}(P_i)(i = 1, 2, \cdots, n)$ and $f = \mathrm{vec}(P)$ are sparse vectors which can be stored in the sparse form. To do so, the memory-cost can be reduced immensely. When the i-th component, x_i, is updated, the coordinate descent method only needs to compute the inner product of some sparse vectors. The coordinate descent method does not involve the matrix-computation, thus has a low computational cost.

3 Convergence analysis

The Lagrange function associated with problem (1.15) is

$$L(x, y, \lambda) = \frac{1}{2}\|Ax - b\|^2 + \frac{1}{2\mu}(e^T x - 1)^2 + \tau\|y\|_{\frac{1}{2}}^{\frac{1}{2}} - \lambda^T(x - y). \tag{3.1}$$

By the first-order optimality conditions, $(x^*, y^*, \lambda^*) \in R^n \times R_+^n \times R^n$ is a stationary point of (1.15) if and only if

$$\begin{cases} A^T(Ax^* - b) + \dfrac{1}{\mu}(e^T x^* - 1)e - \lambda^* = 0, \\ \tau\|y\|_{\frac{1}{2}}^{\frac{1}{2}} - \tau\|y^*\|_{\frac{1}{2}}^{\frac{1}{2}} + (\lambda^*)^T(y - y^*) \geq 0, \quad \forall y \geq 0 \\ x^* - y^* = 0. \end{cases} \tag{3.2}$$

By the optimality conditions of the iteration subproblems (2.2) and (2.3), combining with (2.4) we get

$$\tau(\|y\|_{1/2}^{1/2}) - \tau(\|y^{k+1}\|_{1/2}^{1/2}) + (y - y^{k+1})^T\lambda^{k+1} \geq \rho(y - y^{k+1})^T(x^k - x^{k+1}), \forall\ y \geq 0, \tag{3.3}$$

$$A^T(Ax^{k+1} - b) + \frac{1}{\mu}(e^T x^{k+1} - 1)e - \lambda^{k+1} = r(x^k - x^{k+1}), \tag{3.4}$$

$$x^{k+1} - y^{k+1} = \frac{1}{\rho}(\lambda^k - \lambda^{k+1}). \tag{3.5}$$

Let $v = (x, \lambda)$. It is easy to see that, to prove the convergence to stationary point of the BADMM, one needs only to claim that

$$\lim_{k\to\infty} \|v^{k+1} - v^k\|^2 = 0. \tag{3.6}$$

Lemma 3.1. *For a given $w^k := (x^k, y^k, \lambda^k)$, let $w^{k+1} := (x^{k+1}, y^{k+1}, \lambda^{k+1})$ be generated by BADMM. Then we have*

$$L_\rho(x^{k+1}, y^{k+1}, \lambda^{k+1}) \leq L_\rho(x^k, y^k, \lambda^k) + \frac{1}{\rho}\|\lambda^{k+1} - \lambda^k\|^2 - \frac{r}{2}\|x^{k+1} - x^k\|^2. \tag{3.7}$$

Proof. Since w^{k+1} is generated by the BADMM, by (2.2) and (2.3) we have

$$L_\rho(x^k, y^{k+1}, \lambda^k) \leq L_\rho(x^k, y^k, \lambda^k), \tag{3.8}$$

$$L_\rho(x^{k+1}, y^{k+1}, \lambda^k) \leq L_\rho(x^k, y^{k+1}, \lambda^k) - \frac{r}{2}\|x^{k+1} - x^k\|^2. \tag{3.9}$$

By (2.4) of the BADMM, we get

$$L_\rho(x^{k+1}, y^{k+1}, \lambda^{k+1}) - L_\rho(x^{k+1}, y^{k+1}, \lambda^k) = \frac{1}{\rho}\|\lambda^{k+1} - \lambda^k\|^2. \tag{3.10}$$

Adding (3.8), (3.9) and (3.10) yields (3.7) and completes the proof.

Let $\sigma = \|A^TA + \frac{1}{\mu}ee^T + rI\|$, we have the following lemma.

Lemma 3.2. *For the sequence $\{v^k\}$ generated by the BADMM we have*

$$\|\lambda^{k+1} - \lambda^k\|^2 \leq 2\sigma^2\|x^{k+1} - x^k\|^2 + 2r^2\|x^k - x^{k-1}\|^2. \tag{3.11}$$

Proof. By (3.4), we get

$$\lambda^{k+1} = A^T(Ax^{k+1} - b) + \frac{1}{\mu}(e^Tx^{k+1} - 1)e + r(x^{k+1} - x^k),$$

and at the previous iteration, we have

$$\lambda^k = A^T(Ax^k - b) + \frac{1}{\mu}(e^Tx^k - 1)e + r(x^k - x^{k-1}).$$

Hence

$$\lambda^{k+1} - \lambda^k = (A^TA + \frac{1}{\mu}ee^T + rI)(x^{k+1} - x^k) - r(x^k - x^{k-1}). \tag{3.12}$$

Taking norm on both sides of (3.12) and using triangle inequality, we get

$$\begin{aligned}\|\lambda^{k+1} - \lambda^k\| &\leq \|A^TA + \frac{1}{\mu}ee^T + rI\|\|x^{k+1} - x^k\| + r\|x^k - x^{k-1}\| \\ &= \sigma\|x^{k+1} - x^k\| + r\|x^k - x^{k-1}\|.\end{aligned}$$

By the fact $a^2 + b^2 \geq 2ab$, we have

$$\|\lambda^{k+1} - \lambda^k\|^2 \leq 2\sigma^2\|x^{k+1} - x^k\|^2 + 2r^2\|x^k - x^{k-1}\|^2, \tag{3.13}$$

which completes the proof.

Combing Lemma 3.1 and Lemma 3.2, we get

$$L_\rho(x^{k+1}, y^{k+1}, \lambda^{k+1}) \leq L_\rho(x^k, y^k, \lambda^k) - (\frac{r}{2} - \frac{2\sigma^2}{\rho})\|x^{k+1} - x^k\|^2 + \frac{2r^2}{\rho}\|x^k - x^{k-1}\|^2,$$

which deduces to

$$\begin{aligned}L_\rho(x^{k+1}, y^{k+1}, \lambda^{k+1}) + \frac{2r^2}{\rho}\|x^{k+1} - x^k\|^2 \leq L_\rho(x^k, y^k, \lambda^k) + \frac{2r^2}{\rho}\|x^k - x^{k-1}\|^2 \\ - (\frac{r}{2} - \frac{2(\sigma^2 + r^2)}{\rho})\|x^{k+1} - x^k\|^2.\end{aligned} \tag{3.14}$$

It is easy to show, if the penalty parameter $\rho > 0$ is large enough, the augmented Lagrange function $L_\rho(x, y, \lambda)$ is bounded below. Assume that

$$\rho > \frac{4\sigma^2}{r} + 4r,$$

which implies

$$\kappa = \frac{r}{2} - \frac{2(\sigma^2 + r^2)}{\rho} > 0.$$

Then we have the following theorem.

Theorem 3.1. *The sequence $\{v^k = (x^k, \lambda^k)\}$ generated by the BADMM is asymptotically regular, i.e.,*

$$\lim_{k\to\infty} \|v^{k+1} - v^k\|^2 = 0. \tag{3.15}$$

Proof. By (3.14), we get

$$\begin{aligned}\kappa\|x^{k+1} - x^k\|^2 \le &\Big\{L_\rho(x^k, y^k, \lambda^k) + \frac{2r^2}{\rho}\|x^k - x^{k-1}\|^2\Big\} \\ &- \Big\{L_\rho(x^{k+1}, y^{k+1}, \lambda^{k+1}) + \frac{2r^2}{\rho}\|x^{k+1} - x^k\|^2\Big\}.\end{aligned} \tag{3.16}$$

Summing (3.16) with respect to k yields

$$\begin{aligned}\kappa\sum_{j=1}^{k}\|x^{j+1} - x^j\|^2 \le &\Big\{L_\rho(x^1, y^1, \lambda^1) + \frac{2r^2}{\rho}\|x^1 - x^0\|^2\Big\} \\ &- \Big\{L_\rho(x^{k+1}, y^{k+1}, \lambda^{k+1}) + \frac{2r^2}{\rho}\|x^{k+1} - x^k\|^2\Big\}.\end{aligned} \tag{3.17}$$

Since the augmented Lagrange function $L_\rho(x, y, \lambda)$ is lower-bounded, $L_\rho(x^{k+1}, y^{k+1}, \lambda^{k+1}) + \frac{2r^2}{\rho}\|x^{k+1} - x^k\|^2$ is also lower-bounded for any k. We have

$$\sum_{j=1}^{k}\|x^{j+1} - x^j\|^2 \le C < \infty,$$

which implies

$$\lim_{k\to\infty} \|x^{k+1} - x^k\|^2 = 0. \tag{3.18}$$

Combing this with Lemma 3.2, we have

$$\lim_{k\to\infty} \|\lambda^{k+1} - \lambda^k\|^2 = 0. \tag{3.19}$$

The assertion (3.15) follows directly from (3.18) and (3.19).

Theorem 3.2. *The sequence $\{w^k\}$ generated by the BADMM converges to a stationary point of problem* (1.15).

Proof. Denote $w^\infty := \lim_{k\to\infty} w^k$. Taking limit as $k \to \infty$ on (3.3)-(3.4) and (3.5), and using Theorem 3.1, we have $w^\infty \in R^n \times R^n_+ \times R^n$ and

$$\begin{cases} A^T(Ax^\infty - b) + \dfrac{1}{\mu}(e^T x^\infty - 1)e - \lambda^\infty = 0, \\ \tau\|y\|^{\frac{1}{2}}_{\frac{1}{2}} - \tau\|y^\infty\|^{\frac{1}{2}}_{\frac{1}{2}} + (\lambda^\infty)^T(y - y^\infty) \ge 0, \quad \forall y \ge 0, \\ x^\infty - y^\infty = 0. \end{cases} \tag{3.20}$$

This implies w^∞ is a stationary point of (1.15). Hence, the sequence $\{w^k\}$ generated by the BADMM converges to a stationary point of problem (1.15).

4 Numerical experiments

To demonstrate the performance of the BADMM, some numerical results on the Probabilistic Boolean Network will be presented in this section. All the testing examples are chosen from the BN/PBN MATLAB-based toolbox introduced by Lähdesmäki and Shmulevich [17].

All methods used in these numerical experiments are coded in MATLAB 2014a and run on a personal computer Macbook Pro with 2.6 GHz Intel Core i5 and 8GB RAM.

The stopping criterion of the BADMM is set to $\|v^{k+1} - v^k\|_\infty < \varepsilon$, where $\varepsilon > 0$ is a small real number. The parameters used in the BADMM are set to

$$\rho = 0.8, \mu = 0.3, r = 1.6, \eta = 0.95.$$

For subproblem (2.2), the sparsity parameter s is set by a cross-validation, and the $L_{\frac{1}{2}}$ regularization parameter τ is set by

$$\tau_t = \frac{\sqrt{96}}{9\eta}\rho\|[\theta(y^t)]_{s+1}\|^{\frac{3}{2}}, \tag{4.1}$$

where t is inner iteration counter of Algorithm 2.2. The initial point is set to

$$x_0 = \mathbf{0}, y_0 = \mathbf{1}, \lambda_0 = \mathbf{0}.$$

Example 4.1. There are $j = 2$ genes, and each column of the given transition-probability matrix P has $l = 2$ non-zero entries. The observed transition-probability matrix of the PBN is

$$P = \begin{pmatrix} 0.1 & 0.3 & 0.5 & 0.6 \\ 0.0 & 0.7 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.5 & 0.0 \\ 0.9 & 0.0 & 0.0 & 0.4 \end{pmatrix}$$

In this case, there are 16 Boolean network matrices. The sparse probabilistic Boolean network problem seeks a sparse solution of the following system

$$\sum_{i=1}^{16} x_i P_i = P \tag{4.2}$$

with $x \geq 0$ and $e^T x = 1$.

For this example, three methods are used: the BADMM proposed in this paper, the maximum entropy rate approach (MEM) proposed by Chen, Ching and Chen [1] and the modified maximum entropy rate approach (MMEM) proposed by Chen, Jiang and Ching [2]. The computational results are displayed in Figure 1 for easy comparison. The solution generated by the BADMM is as sparse as that of the MMEM, and both are sparser than the solution generated by the MEM. We have noticed that, the MMEM also emphasized the sparsity by using L_α-regularization. To do so, the objective function of the MMEM is

$$\max_x \left\{ -\sum_{i=1}^{n} x_i \log x_i - \beta \sum_{i=1}^{n} x_i^\alpha \right\}. \tag{4.3}$$

With the best choice of the parameter pair (α, β), the MMEM obtains a sparse solution of the PBN problem. The computational results of the MMEM are very sensitive to the parameter pair (α, β). In the case of that there are 2 genes and each column of the given transition- probability matrix P has 2 non-zero entries, the best choice of the parameter pair $(\alpha, \beta) = (0.63, 1.40)$ is determined by numerical experiments without theoretical principle. However, the BADMM is independent of experimental parameters, and also gives a solution as sparse as that of the MMEM. However, the model of the MEM is a convex optimization, the solution generated by MEM is more logical since it is a global optimal solution. The MEM identifies six major Boolean networks, they are $P_9, P_{11}, P_{13}, P_{14}, P_{15}$ and P_{16}, and the BADMM gives a more sparse solution with four major Boolean networks. The MMEM gives a different solution, and the identified major Boolean networks are $P_6, P_8, P_{10}, P_{12}, P_{13}$ and P_{15}.

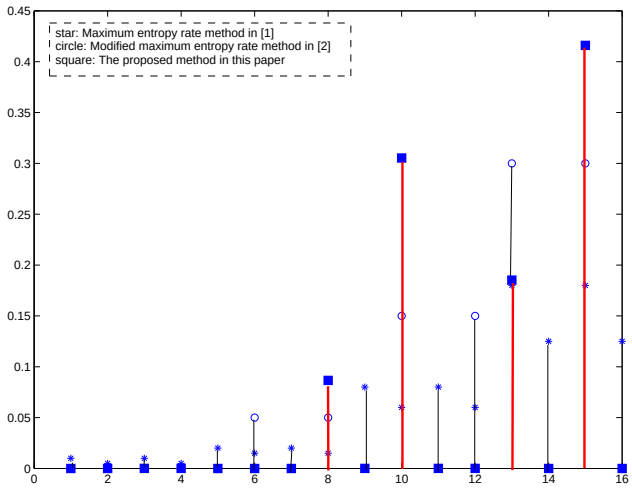

Figure 1: The probability distribution x for the case $j = 2$ and $l = 2$

Example 4.2. There are $j = 2$ genes, and each column of the observed transition-probability matrix P has $l = 3$ non-zero entries, where

$$P = \begin{pmatrix} 0.1 & 0.3 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.2 & 0.0 \\ 0.0 & 0.0 & 0.6 & 0.4 \\ 0.7 & 0.4 & 0.0 & 0.5 \end{pmatrix}.$$

In this case, there are 81 Boolean network matrices.

The MMEM obtains the best choice of the parameter pair $(0.61, 0.6)$ from 1980 pairs of (α, β), then it gives a sparse solution identifying six major Boolean networks. The BADMM identifies eight major Boolean networks, and the re-constructed PBN is dominated by the $5^{th}, 11^{th}, 47^{th}, 50^{th}, 63^{rd}, 72^{nd}, 73^{rd}$ and 80^{th} BNs. The numerical solutions generated by the three methods are displayed in Figure 2 for comparison. Compared with results of the MEM and MMEM, the solution generated by the BADMM is more sparse, and the BADMM is able to more readily identify the major BNs in the PBN.

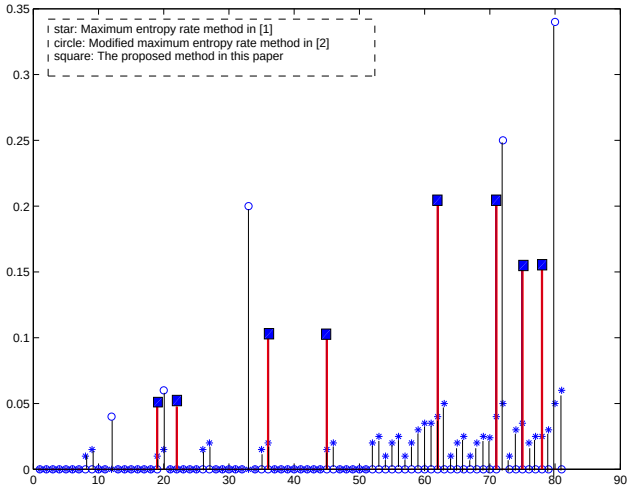

Figure 2: The probability distribution x for the case $j = 2$ and $l = 3$

Example 4.3. There are 1024 BNs, and each column of the observed transition-probability matrix P has different non-zero entries, where

$$P = \begin{pmatrix} 0.12 & 0 & 0.60 & 0.42 & 0 & 0 & 0 & 0 \\ 0.28 & 0 & 0 & 0.18 & 0 & 0 & 0 & 0 \\ 0 & 0.40 & 0 & 0 & 0.40 & 0.18 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.42 & 0 & 0.60 \\ 0.18 & 0 & 0.40 & 0.28 & 0 & 0 & 0 & 0 \\ 0.42 & 0 & 0 & 0.12 & & 0 & 0 & 0 \\ 0 & 0.60 & 0 & 0 & 0.60 & 0.12 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.28 & 1.00 & 0.40 \end{pmatrix}$$

Fixing the other parameters, we obverse the effect of stopping error $\varepsilon > 0$ to the computational results. The computational results generated by the BADMM are put in Table 1. From Table 1, we find that the identified major BNs keep unchanged while the computational cost increases as the stopping error ε is deceasing continuously to 1.0×10^{-5}. This fact implies that, the identified major BNs generated by the BADMM is spot-on, reliable, and robust under the different error setting.

When the stopping error is set to $\varepsilon = 1.0 \times 10^{-3}$, the BADMM identifies 7 major BNs from 1024 BNs with 260 iterations. The reconstructed transition probability matrix is

$$\hat{P} = \begin{pmatrix} 0.1199 & 0 & 0.5999 & 0.42 & 0 & 0 & 0 & 0 \\ 0.2801 & 0 & 0 & 0.18 & 0 & 0 & 0 & 0 \\ 0 & 0.4001 & 0 & 0 & 0.4001 & 0.18 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.42 & 0 & 0.5999 \\ 0.18 & 0 & 0.4001 & 0.2801 & 0 & 0 & 0 & 0 \\ 0.42 & 0 & 0 & 0.1199 & 0 & 0 & 0 & 0 \\ 0 & 0.5999 & 0 & 0 & 0.5999 & 0.1199 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.28 & 1.00 & 0.4001 \end{pmatrix}.$$

Table 1: The computational results of BADMM with different stopping error ε

Stopping error ε	10^{-2}	10^{-3}	5×10^{-4}	10^{-4}	5×10^{-5}	10^{-5}
Total iteration number k	97	260	267	520	562	722
Identified major BNs	104					
	118					
	189	118	118	118	118	118
	358	360	360	360	360	360
	360	395	395	395	395	395
	376	594	594	594	594	594
	395	836	836	836	836	836
	594	911	911	911	911	911
	836	939	939	939	939	939
	911					
	939					

The reconstructed residual is $\|\hat{P}-P\|_F = 3.4067\times10^{-4}$, and the cpu-time cost is 66.86 seconds. The major BNs identified by the BADMM is displayed in Figure 3.

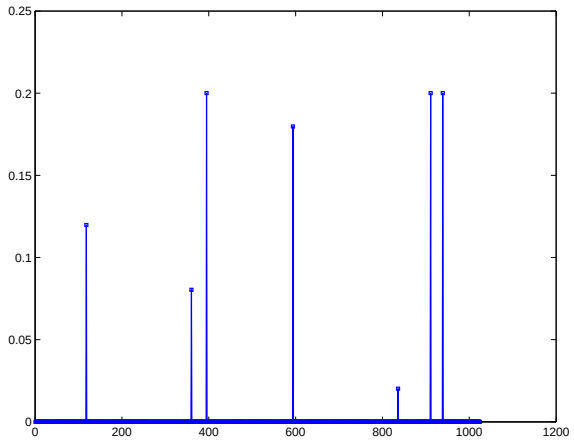

Figure 3: The probability distribution x for the case $j = 3$ and 1024 BNs

Example 4.4. There are $j = 3$ genes, and each column of the observed transition-probability

matrix P has 2 or 4 non-zero entries, where

$$P = \begin{pmatrix} 0.5672 & 0.4328 & 0.2881 & 0 & 0.1447 & 0 & 0.4328 & 0 \\ 0 & 0 & 0.1447 & 0 & 0.2881 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.3776 \\ 0 & 0 & 0 & 0.4328 & 0 & 0 & 0 & 0.1896 \\ 0.4328 & 0.5672 & 0.3376 & 0 & 0.1896 & 0 & 0.5672 & 0 \\ 0 & 0 & 0.1896 & 0 & 0.3776 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.6657 & 0 & 0.2881 \\ 0 & 0 & 0 & 0.5672 & 0 & 0.3343 & 0 & 0.1447 \end{pmatrix}.$$

In this case, there are 2048 BNs.

By setting the stopping error $\varepsilon = 1.0 \times 10^{-3}$ and the sparsity parameter $s = 20$, the BADMM identifies 14 major BNs from 2048 BNs within 307 iterations. The reconstructed residual $\|\hat{P} - P\|_F = 5.1609 \times 10^{-4}$ and the cpu-time cost is 524.21 seconds. The identified major BNs are displayed in Figure 4.

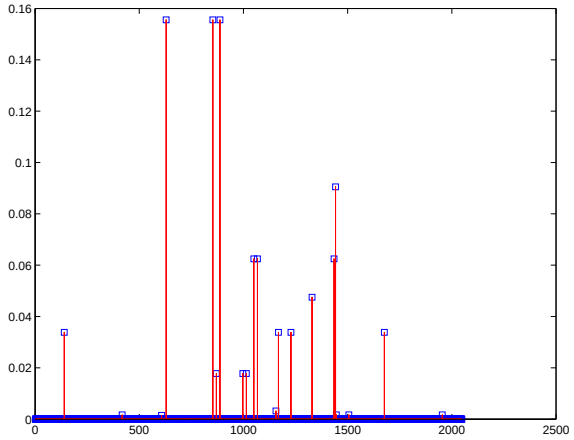

Figure 4: The probability distribution x for the case $j = 3$ and 2048 BNs

Example 4.5. There are $j = 4$ genes, and each column of the observed transition-probability matrix P has 1 or 2 non-zero entries. The matrix P to be reconstructed has the sparse form as follows:

P(1,1)=1.0	P(1,4)=1.0	P(1,8)=0.5078	P(1,12)=0.5078
P(2, 3)=0.5078	P(2,7)=0.5078	P(2, 11)=0.5078	P(2,15)=0.5078
P(5, 2)=1.0	P(5, 6)=1.0	P(5,10)=0.5078	P(5,14)=0.5078
P(6, 4)=0.5078	P(6,8)=0.5078	P(6,12)=0.5078	P(6,16)=0.5078
P(9, 9)=0.4922	P(9,13)=0.4922	P(10,3)=0.4922	P(10,7)=0.4922
P(10,11)=0.4922	P(10,15)=0.4922	P(13, 10)=0.4922	P(13, 14)=0.4922
P(14, 4)=0.4922	P(14, 8)=0.4922	P(14, 12)=0.4922	P(14, 16)= 0.4922

In this case, there are 4096 BNs.

By setting the stopping error $\varepsilon = 1.0 \times 10^{-3}$ and the sparsity parameter $s = 40$, the BADMM identifies 79 major BNs from 4096 BNs within 44 iterations. The reconstructed residual is $\|\hat{P} - P\|_F = 1.2 \times 10^{-4}$, and the cpu-time cost is 426.85 seconds. The identified major BNs are displayed in Figure 5.

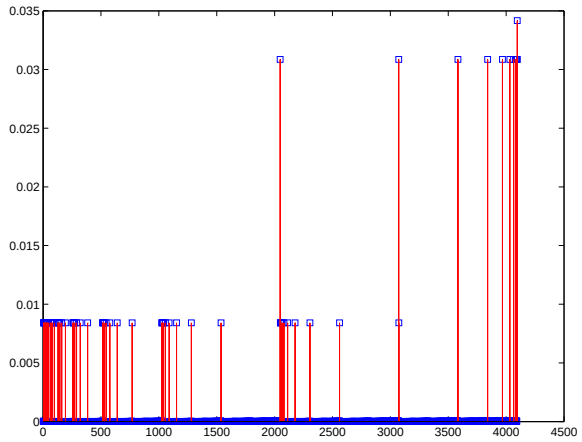

Figure 5: The probability distribution x for the case $j = 4$ and 4096 BNs

In summary, the numerical results show that, the proposed BADMM is valid for solving problem (1.15) resulted from the sparse probabilistic Boolean network model (1.13). Compared with the MEM and MMEM, model (1.13) and the BADMM proposed in this paper are more steady and efficient for the sparse probability Boolean network reconstruction problem.

5 Conclusions

The genetic regulatory networks are very useful in molecular biology of gene and genetic biology, e.g., gene diagnosis. From the view of genetic diagnosis point, a virulence gene is driven from some interrelated normal genes. Probability Boolean network models the interrelated normal genes as a set of Boolean networks, and the virulence gene as a transition-probability matrix. The gene diagnosis is devoted to finding a sparse probabilistic Boolean network, which is used to distinguish the major genes which resulting to the virulence gene.

This paper reformulated the sparse probability Boolean networks problem to a separable $L_{\frac{1}{2}}$ regularization optimization problem, and proposed a Bregman alternating direction method of multipliers (BADMM) for resulting optimization problem. Under some mild conditions, the convergence to stationary point of the BADMM was established.

By the customized subproblem-solvers of the proposed BADMM, all the iteration subproblems are computationally tractable, and consequently the proposed method is efficient. Numerical results indicate that, the BADMM is numerically steady and efficient for the sparse probabilistic Boolean networks problem.

The L_0 regularization optimization model (1.5), and a variant of the form

$$\begin{cases} \min\limits_{x\in R^n} \dfrac{1}{2}\Big\|Ax-b\|^2, \\ \text{s.t. } \|x\|_0 \le \kappa,\ \mathbf{e}^T x = 1,\ x \ge 0, \end{cases} \tag{5.1}$$

has some important applications in the genetic regulatory networks as well as sparse probabilistic Boolean network. The recent development of optimization technique, such as projected gradient methods proposed by Xu, Dai, et al [18], provides some useful approaches to solve problems (1.5) and (5.1) with high efficiency. However, at information representation, the regularization parameter τ in (1.5) and the sparsity parameter κ in (5.1) are sensitive to practical applications of the sparse probabilistic Boolean network. Hence, model (1.5) and its variant (5.1), and the corresponding computational algorithms, are the future works of our group.

Acknowledgments This work was supported by Natural Science Foundation of China (Grant No. 11571074), Natural Science Foundation of FuJian Province (Grant No. 2015J01010). The first version of this work was completed during the second author visiting Professor Dai Y-H at Academy of Mathematics and Systems Science, Chinese Academy of Sciences. The authors are grateful to Professor Han D-R at Nanjing Normal University and Professor Yuan X-M at Hong Kong Baptist University for some helpful discussions.

References

[1] Chen X, Ching W K and Chen, X S, et al. Construction of Probabilistic Boolean Networks from a Prescribed Transition Probability Matrix: A Maximum Entropy Rate Approach, East Asian Journal on Applied Mathematics, 2011, 1(2): 132-154.

[2] Chen X, Jiang H andChing W. K. Construction of Sparse Probabilistic Boolean Networks, East Asian Journal of Applied Mathematics, 2012, 2:, 1-18.

[3] Shmulevich I, Dougherty E R, Kim S and Zhang W. Probabilistic Boolean networks: a rule- based uncertainty model for gene regulatory networks, Bioinformatics, 2002, 18(2): 261-274.

[4] Shmulevich I, Dougherty E R, and Zhang W. From Boolean to probabilistic Boolean networks as models of genetic regulatory networks. Proceedings of the IEEE, 2002, 90(11): 1778-1792.

[5] Shmulevich I, Dougherty E R and Zhang W. Gene perturbation and intervention in probabilistic Boolean networks. Bioinformatics, 2002, 18(10): 1319-1331.

[6] Natraajan B K. Sparse approximation to linear systems. SIAM Journal on Computing, 1995, 24: 227-234.

[7] Peng Z, Wu D H. A partial parallel splitting augmented Lagrangian method for solving constrained matrix optimization problems, Computers and Mathematics with Applications, 2010, 60: 1515-1524.

[8] He B S, Tao M, Yuan X M. Alternating direction method with Gaussian back substitution for separable convex programming, SIAM Journal on Optimization,2012, 22: 313-340.

[9] Boyd S, Parikh N, Chu E, Peleato B, Eckstein J. Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine Learning, 2010, 3(1): 1-122.

[10] He B S, Yang H. Some convergence properties of a method of multipliers for linearly constrained monotone variational inequalities, Operations Research Letters, 1998, 23: 151-161.

[11] Eckstein J, Fukushima M. Some reformulations and applications of the alternating direction method of multipliers, Large Scale Optimization: State of the Art, Springer US, 1994, 115-134.

[12] Xu M H. Proximal alternating directions method for structured variational inequalities, Journal of Optimization Theory and Applications, 2007, 134: 107-117.

[13] Fukushima M. Application of the alternating direction method of multipliers to separable convex programming problems, Computational Optimization and Applications, 1992, 1(1): 93-111.

[14] Xu Z B, Chang X Y, Xu F M, Zhang H. $L_{\frac{1}{2}}$ regularization: a thresholding representation theory and a fast slover, IEEE Transactions on neural networks and learning systems, 2012, 23(7): 1013-1027.

[15] Xu Z B, Guo H, Wang Y, Zhang H. The representation of $L_{\frac{1}{2}}$ regularizer among $L_q(0 < q < 1)$ regularizer: an experimental study based on phase diagram, Acta Automatica Sinica, 2012, 38(7): 1225-1228..

[16] Zeng J, Lin S, Wang Y, Xu Z B. $L_{\frac{1}{2}}$ Regularization: convergence of iterative half thresholding algorithm. IEEE Transactions on Signal Processing, 2014, 62(9): 2317-2329.

[17] BN/PBN Toolbox. [http://code.google.com/p/pbn-matlab-toolbox]

[18] Xu F M, Dai Y H, Zhao Z H, Xu Z B. Efficient Projected Gradient Methods for A Class of L_0 Constrained Optimization Problems, Mathematical Programming, to appear.